REMARKS/ARGUMENTS

I. Status of Claims

- Claims 1 and 16 are Independent Claims.
- Claims 1-27 remain pending for review.
- Claims 1-10, 12-24, and 26-27 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Bommareddy et al (US Pat. No. 6,880,089) (hereinafter referred to as **Bommareddy**) in view of Tremain (US Publ. No. 2002/0069369 A1) (hereinafter referred to as **Tremain**).
- Claims 11 and 25 are rejected under 35 U.S.C. § 103(a) as being unpatentable over **Bommareddy** in view of **Tremain** and further in view of Bunker, V et al. (US Publ. No. 2003/0028803) (hereinafter referred to as **Bunker, V**).

II. Response

A. A Bommareddy-Tremain combination does not replicate Applicants' claimed invention.

Examiner stated that **Bommareddy** teaches all of Applicants' elements in Independent Claims 1 and 16 except as to where the active system is automatically assumed to have failed, regardless of whether the active subsystem has actually failed. See Office Action (dated 12/12/2008), p. 3, para. 6. However, Examiner believes **Tremain** does.

As a friendly reminder, **Bommareddy** actively looks for firewall failures and/or detects firewall failures by monitoring the operational health of routers. Once a failure is detected, then and only then would **Bommareddy's** firewall clustering system acts. See Amendment (dated 09/12/2008), § II. Examiner Interview Summary; see also Amendment (dated 02/19/2008), § II.A.; Amendment (dated 08/31/2007), § II.A.

As argued in the previous amendments and the Examiner Interview, Applicants have explained why **Bommareddy** does not teach Applicants' claimed invention. In summary, **Bommareddy's** firewall clustering system monitors the operational health of firewalls. See **Bommareddy**, Fig. 8, col. 3, ll. 35-37, col. 7, ll. 23-26 and col. 8, ll. 20-22. In particular, **Bommareddy** actively looks for firewall failures by using the operational health monitoring feature (i.e., testing the operational state of the firewall with Ping packets). Id. at col. 8, ll. 20-26.

To ensure that the servers are operational, these network flow controllers implement server fault-intolerance within a cluster. See **Bommareddy**, col. 21, ll. 17-20. At regular intervals, the network flow controllers would ping each server with application probes and await a reply. Id. at col. 8, ll. 20-26 and col. 21, ll. 16-19. If a server fails to respond, then **Bommareddy** classifies the failed server as being "down." Id. at col. 21, ll. 21-24. If and when a server is down, then the network flow controllers would reroute any packets bound for the down server to the most suitable servers within the cluster. Id. at col. 21, ll. 25-31.

With respect to **Tremain**, **Tremain** sets up, on a real computer, virtual machines with a virtual firewall connection with an external network to provide services to multiple users. See **Tremain**, Abstract, para. [0048]. Part of **Tremain** includes a detection device "for detecting evidence of malicious software or hostile attack signatures". Id. at para. [0050]. Furthermore, **Tremain** also discloses security intrusion detection. Id. at para. [0136].

However, a closer look at **Tremain's** security intrusion detection reveals that **Tremain** does not teach Applicants' claimed automatic cleansing element. First, the name says it all – "Security Intrusion Detection" in the ordinary sense means the computer system found out that

someone broke into the system or that there was attempted or successful, unauthorized access. See **Tremain**, para. [0136].

Second, to provide for security intrusion detection, **Tremain** uses an intrusion detection software. After an intrusion is detected, the virtual intrusion detection system can alert system operators. See **Tremain**, para. [0137]. Alternatively, after an intrusion is detected, the virtual intrusion detection system can automatically shut the system down. Id. Then, the computer can scan the memory and storage space for patterns or the presence of malicious software or activity. Id.

Tremain's teachings present two key points. One, **Tremain** can automatically shut down a system *after* an intrusion is detected. Thus, the point here is that intrusion is a prerequisite. Some intrusion must happen first and be detected before anything else can happen. Two, **Tremain's** automatic shut down does not mean cleansing the system. Rather, **Tremain** specifically said that it looks for the presence of malicious software or activity. If found, then **Tremain** would turn off the system.

When one combines **Bommareddy** with **Tremain**, a **Bommareddy-Tremain** combination may look like the following: a system that actively looks for/detects firewall failures by monitoring the operational health of routers, and upon detecting a security intrusion, automatically scanning the memory and storage space for patterns which indicate the presence of malicious software or activity.

In sharp contrast, this combination is clearly different from Applicants' claimed invention. Applicants do not disclose detecting a system failure or security intrusion. Applicants do not wait for (or even induce) a failure. Furthermore, when Applicants assume that a system failure exits, this assumption does not necessarily mean that a failure or intrusion actually exists.

Rather, Applicants' claimed invention just cleanses one or more subsystems on a cyclical basis regardless of whether failure or intrusion has occurred. In other words, Applicants' self-cleansing cycles are independent of failure occurrences or detection of security intrusions.

As previously presented, Applicants' self-cleansing mechanism enters into an automatic cleansing process by automatically cleansing at least one subsystem (e.g., a firewall, server, gateway, etc.) on a cyclically timed-basis. See, e.g., Amendment (dated 02/19/2008), § II.A.; see also **Specification**, Figs. 1-5, paras. [0023]-[0024], [0038]. These self-cleansing activities will occur regardless of whether a fault in an active subsystem is detected or an intrusion into an active subsystem is detected. See **Specification**, para. [0040]. As such, even if an intrusion was successful, the intrusion would be limited to a very short window of one fast, self-cleansing cycle. Id. at paras. [0043], [0044] and [0057].

The main point to take away from this important distinction is that failure or an intrusion actually exists in **Bommareddy** and **Tremain**. **Bommareddy** is dependent upon and must look for those failures in order for the invention to work. Similarly, **Tremain** must first detect that a security intrusion has occurred prior to automatically scanning the memory for malicious software or activity. For **Bommareddy** and **Tremain** to work, the failure/security intrusion prerequisite must first be satisfied. Otherwise, no scanning will take place in either **Bommareddy** or **Tremain**.

In contrast, Applicants do not have this prerequisite. Hence, this lacking makes it very clear that the **Bommareddy-Tremain** combination does not replicate Applicants' self-cleansing model. Because of these substantial differences, the arguments presented here overcome the § 103(a) rejections. Thus, Applicants respectfully request Examiner to withdraw these rejections.

B. Because the Bommareddy-Tremain combination teaches an art that is substantially different from Applicants' claimed invention, a Bommareddy-Tremain-Bunker, V combination does not replicate Dependent Claims 11 and 25.

Examiner cited **Bunker, V** in combination with **Bommareddy** and **Tremain** to teach the auditing limitations of the claimed invention because neither **Bommareddy** nor **Tremain** teaches the auditing limitations. See Office Action (dated 12/12/2008), p. 7, paras. 2-3. In particular, **Bunker, V** discloses assessing the vulnerability of an internal network. See **Bunker, V**, paras. [0115] and [0121].

Although **Bunker, V** does disclose such assessment, this disclosure does not aid the **Bommareddy-Tremain** combination to reject Applicants' claimed invention based on obviousness. **Bommareddy-Tremain** still remains a system combination that actively looks for/detects firewall failures by monitoring the operational health of routers, and upon detecting a security intrusion, automatically scans the memory and storage space for patterns which indicate the presence of malicious software or activity. See supra § II.A. As one can see, adding **Bunker, V's** auditing disclosure does not help create a combination that teaches Applicants' self-cleansing system that automatically cleanses the system without first detecting a system failure or intrusion. Rather, if **Bunker, V** were combined with **Bommareddy-Tremain**, **Bunker, V** would merely add the auditing limitation to the **Bommareddy-Tremain** combination. Such **Bommareddy-Tremain-Bunker, V** combination bears no resemblance to Applicants' overall claimed invention.

In contrast, the auditing limitation Applicants teach is directed to system functions, such as self-cleansing, measureable events, and system performance. See **Specification**, para. [0034].

As Applicants' auditing feature monitors the self-cleansing cycle, it can record events for further analysis and archiving. Id.

Because combining **Bunker, V** with **Bommareddy-Tremain** does not help in replicating Applicants' claimed invention, Applicants respectfully request Examiner to withdraw these § 103(a) rejections.

C. Dependent Claims 2-15 and 17-27 depend on Independent Claims.

Because Dependent Claims 2-15 and 17-27 ultimately depend on their respective independent claims, the arguments presented for the independent claims also apply to these dependent claims. Therefore, Applicants respectfully request withdrawal of these objections.

III. Conclusion

For all of the reasons advanced above, Applicants respectfully believe that the application overcomes Examiner's 35 U.S.C. § 103 concerns. If there are any outstanding issues that might be resolved by an interview or an Examiner's Amendment, Applicants request that the Examiner call the Applicants' agents at the telephone number shown below.

IV. Deposit Account

Applicants hereby authorize the Commissioner to credit or debit any outstanding fees in connection with this patent application using Deposit Account No. 50-3212.

Respectfully submitted,

/David Yee, Reg. No. 55,753/
David Yee, Registration No. 55,753

Filed: February 27, 2009

Office of Technology Transfer,
George Mason University
4400 University Dr., MSN5G5
Fairfax, VA 22030
Phone: 703-993-3949

Fax: 703-993-9710
E-mail: dyee@gmu.edu